



VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA  
EKONOMICKÁ FAKULTA

KATEDRA APLIKOVANÉ INFORMATIKY

Návrh a implementace databázové aplikace pro sběrnou druhotných surovin  
Design and Implementation of Database Application for the Salvaging Company

Student: David Krčmář  
Vedoucí bakalářské práce: Ing. Vítězslav Novák, Ph.D.

Ostrava 2017

VŠB - Technická univerzita Ostrava  
Ekonomická fakulta  
Katedra aplikované informatiky

## Zadání bakalářské práce

Student:

**David Krčmář**

Studijní program:

B6209 Systémové inženýrství a informatika

Studijní obor:

6209R017 Informatika v ekonomice

Téma:

Návrh a implementace databázové aplikace pro sběru druhotných  
surovin

Design and Implementation of Database Application for the Salvaging  
Company

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Úvod
  2. Teoretické pojmy a postupy tvorby databází
  3. Současná situace ve firmě
  4. Návrh a implementace databázové aplikace
  5. Závěr
- Seznam použité literatury  
Seznam zkratk  
Prohlášení o využití výsledků bakalářské práce  
Seznam příloh  
Přílohy

Seznam doporučené odborné literatury:

PÍSEK, Slavoj. *Access 2013: podrobný průvodce*. Praha: Grada Publishing, 2013.  
ISBN 978-80-247-4746-0.

CONNOLLY, T. M., C. E. BEGG a R. HOLOWCZAK. *Mistrovství - databáze: profesionální průvodce  
tvorbou efektivních databází*. Brno: Computer Press, 2009. ISBN 978-80-251-2328-7.

SHEPHER, Richard. *Access VBA: výukový průvodce*. Brno: Computer Press, 2012.  
ISBN 978-80-251-3686-7.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Vítězslav Novák, Ph.D.**

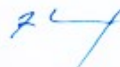
Datum zadání: 18.11.2016

Datum odevzdání: 05.05.2017



---

Ing. Petr Rozehnal, Ph.D.  
vedoucí katedry



---

prof. Dr. Ing. Zdeněk Zmeškal  
děkan fakulty

### **Prohlášení studenta**

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

Zároveň bych na tomto místě chtěl poděkovat svému vedoucímu bakalářské práce, Ing. Vítězslavu Novákovi, PhD. za vedení bakalářské práce, cenné rady a připomínky.

V Ostravě dne 3. 5. 2017



.....

podpis studenta

# Obsah

<b>1</b>	<b>Úvod .....</b>	<b>6</b>
<b>2</b>	<b>Teoretické pojmy a postupy tvorby databází .....</b>	<b>8</b>
2.1	Základní teoretické pojmy.....	8
2.1.1	Databáze .....	8
2.1.2	Data .....	8
2.1.3	Informace .....	8
2.1.4	Metadata.....	8
2.1.5	Databázová aplikace.....	9
2.1.6	Databázový systém .....	9
2.2	Vrstvy datové abstrakce.....	10
2.2.1	Fyzická vrstva .....	11
2.2.2	Logická vrstva .....	11
2.2.3	Externí vrstva.....	11
2.3	Databázový model.....	11
2.3.1	Hierarchický model .....	12
2.3.2	Síťový model.....	13
2.3.3	Relační model.....	13
2.3.4	Objektově orientovaný model .....	14
2.3.5	Objektově relační model .....	15
2.3.6	NoSQL databáze .....	15
2.4	Datové modelování .....	15
2.4.1	Entity .....	15
2.4.2	Atributy .....	16
2.4.3	Relace .....	17
a)	Relace 1:1 .....	17
b)	Relace 1:N .....	17
c)	Relace M:N .....	18
2.4.4	Klíče relace .....	19
a)	Superklíč .....	19
b)	Kandidátní klíč .....	19
c)	Primární klíč.....	19
d)	Cizí klíč.....	19
2.5	Normalizace .....	19

2.5.1	První normální forma .....	20
2.5.2	Druhá normální forma.....	21
2.5.3	Třetí normální forma .....	21
2.5.4	Boyce-Coddova normální forma .....	22
2.5.5	Čtvrtá a pátá normální forma.....	22
2.6	Transakce .....	22
2.7	Microsoft Access .....	23
2.7.1	Datové typy .....	23
2.7.2	Tabulky .....	24
2.7.3	Dotazy .....	24
2.7.4	Formuláře.....	24
2.7.5	Sestavy .....	25
2.7.6	Makra .....	25
2.7.7	Moduly .....	25
2.8	Zákony a vyhlášky pro sběrné suroviny .....	26
2.8.1	Katalog odpadů .....	26
2.8.2	Zákon o odpadech.....	27
<b>3</b>	<b>Současná situace ve firmě.....</b>	<b>29</b>
3.1	Charakteristika firmy.....	29
3.2	Analýza současné situace.....	29
3.3	Požadavky na databázovou aplikaci.....	30
<b>4</b>	<b>Návrh a implementace databázové aplikace.....</b>	<b>31</b>
4.1	Návrh databáze .....	32
4.2	Rozbor použitých tabulek.....	33
4.2.1	Tabulka Dodavatele.....	33
4.2.2	Tabulka Odberatele.....	35
4.2.3	Tabulka Sklady.....	36
4.2.4	Tabulka Karta_odpadu .....	37
4.2.5	Tabulka Pohyb transakcí .....	38
4.3	Rozhraní databázové aplikace.....	39
4.3.1	Formulář frmDomu .....	39
4.3.2	Formulář frmDodavatele.....	40
4.3.3	Formulář frmSklady.....	42

4.3.4	Formulář frmKartaOdpadu.....	44
4.3.5	Formulář frmPohybTransakci.....	45
4.3.6	Formulář frmOdberatele.....	47
4.3.7	Závěrečné přizpůsobení pro uživatele .....	48
<b>5</b>	<b>Závěr.....</b>	<b>49</b>
	<b>Seznam použité literatury .....</b>	<b>50</b>
	<b>Seznam zkratk.....</b>	<b>52</b>
	<b>Prohlášení o využití bakalářské práce</b>	
	<b>Seznam příloh</b>	



# 1 Úvod

Tématem bakalářské práce je návrh a následná implementace databázové aplikace v programu MS Access 2016, kterou využívá sběrna druhotných surovin.

Databáze je nástroj, který slouží pro shromažďování a uspořádávání informací. Mohou v ní být uloženy informace o osobách, zboží, materiálu, objednávkách a čemkoliv jiném, co je potřeba. Poté se tyto informace mohou využít ve firmách, kde jsou prospěšné například při správě zboží nebo ke kontrole množství.

Sběrna, pro niž je databáze sestavena, se soustředí na výkup a následný prodej kovů (hlavně mědi, oceli, železa, hliníku) a papíru. Tyto suroviny jsou před prodejem zpracovány a roztříděny podle jednotlivých druhů kovů a papíru. Aby sběrna měla lepší přehled o všech surovinách, které má na skladě a poté je mohla výhodně prodat, databázová aplikace bude zobrazovat množství a také jednotkovou cenu jednotlivých surovin. Důležitým aspektem aplikace je databáze všech odběratelů, kterým může dané suroviny dodávat. Jelikož sběrna dosud žádnou databázi nedisponuje, tak je daná databáze vytvořena s důrazem na přehlednost a potřeby sběrní.

Cílem této bakalářské práce je vytvořit databázovou aplikaci, která splňuje následující funkce:

- jednoduché rozhraní a intuitivní ovládání,
- správa a záznam pohybu veškerého kovu a papíru,
- seznam všech údajů o dodavatelích a odběratelích,
- záznam veškerých skladů sběrní.

Obsah bakalářské práce je rozdělen do pěti základních kapitol, z nichž první je samotný *Úvod*, ve kterém je stručně popsána problematika a jednotlivé cíle této bakalářské práce.

Ve druhé části s názvem *Teoretické pojmy a postupy tvorby databází* se práce zabývá zejména teorií, která je důležitá při tvorbě a fungování databázové aplikace.

V kapitole *Současná situace ve firmě* je vysvětleno, jak bude aplikace sběrně druhotných surovin prospěšná a v čem bude spočívat její přínos.

Předposlední část *Návrh a implementace databázové aplikace* je zaměřena na praktickou stránku aplikace. Jsou zde popsány relace mezi jednotlivými tabulkami. Následně jsou uvedeny formuláře a tiskové sestavy, které jsou využívány.

V *Závěru* je popsáno zhodnocení této bakalářské práce a jejích jednotlivých cílů.

## **2 Teoretické pojmy a postupy tvorby databází**

Obsahem této kapitoly jsou pojmy a postupy, které jsou použity při tvorbě databázové aplikace. Popis základních informací a seznámení s programem Microsoft Access. Dále jsou zde popsány také vyhlášky a pravidla, kterými se musí řídit každá sběrna druhotných surovin, která provozuje svou činnost v České republice.

### **2.1 Základní teoretické pojmy**

Tato podkapitola se zabývá vysvětlením základních pojmů, které jsou důležité a je nutné je znát při vytváření jakékoliv databázové aplikace

#### **2.1.1 Databáze**

Databáze je kolekce dat, které spolu vzájemně souvisejí a pracujeme s ní jako s ucelenou jednotkou, která obsahuje jediné úložiště dat, a to může současně využívat mnoho uživatelů. Veškerá data, která jsou uživateli požadovaná, bývají integrovaná s minimálním množstvím duplikací. Databáze také obsahuje provozní data spolu s popisem těchto dat. (Oppel, 2006; Conolly, Begg a Holowczak, 2009)

#### **2.1.2 Data**

Veškeré uložené údaje v databázi se označují jako data. Jsou to surová fakta, která jsou pro určité lidi důležitá. Data se musejí poté dále zpracovat, aby z nich bylo možné získat určité informace. (Conolly, Begg a Holowczak, 2009)

#### **2.1.3 Informace**

Informace jsou data, která již prošla určitým zpracováním nebo dostala strukturu a lidem dávají určitý význam. (Conolly, Begg a Holowczak, 2009)

#### **2.1.4 Metadata**

Metadata jsou nejjednodušeji definována jako data o datech. Existuje také jiný název, kterým jsou označována a tím je systémový katalog. Metadata jsou data, která popisují objekty v databázi a díky nim je snazší manipulovat s těmito daty. Popisují strukturu databáze, zabezpečení a velikost datových typů. (Singh, 2011)

### **2.1.5 Databázová aplikace**

Databázová aplikace je počítačový program, který je propojený s určitou databází vyvoláním odpovídajícího požadavku. Požadavky se tvoří pomocí jednoho nebo více příkazů SQL. Uživatelé většinou propojují databázi prostřednictvím více databázových aplikací, ty se poté využívají ke správě, vytvoření a generování informací z databáze. Databázové aplikace se vytvářejí jako dávkové aplikace nebo, v dnešní době spíše častěji, jako online aplikace. Tyto aplikace lze dnes programovat v programovacích jazycích třetí generace jako je C++, Java, VBA a C#, který poté posílá tyto požadavky na data do databázového systému nejčastěji pomocí příkazů SQL jazyka. (Conolly, Begg a Holowczak, 2009)

### **2.1.6 Databázový systém**

Databázový systém v české literatuře často označovaný jako Systém řízení báze dat je software, který je dodáváný vybraným výrobcem, ke správě databáze. Příklady databázových systémů jsou Microsoft Access, Oracle, MySQL a DB2 od firmy IBM. (Oppel, 2006)

Databázový systém zajišťuje základní služby, které jsou nezbytné pro organizaci databáze, jako například:

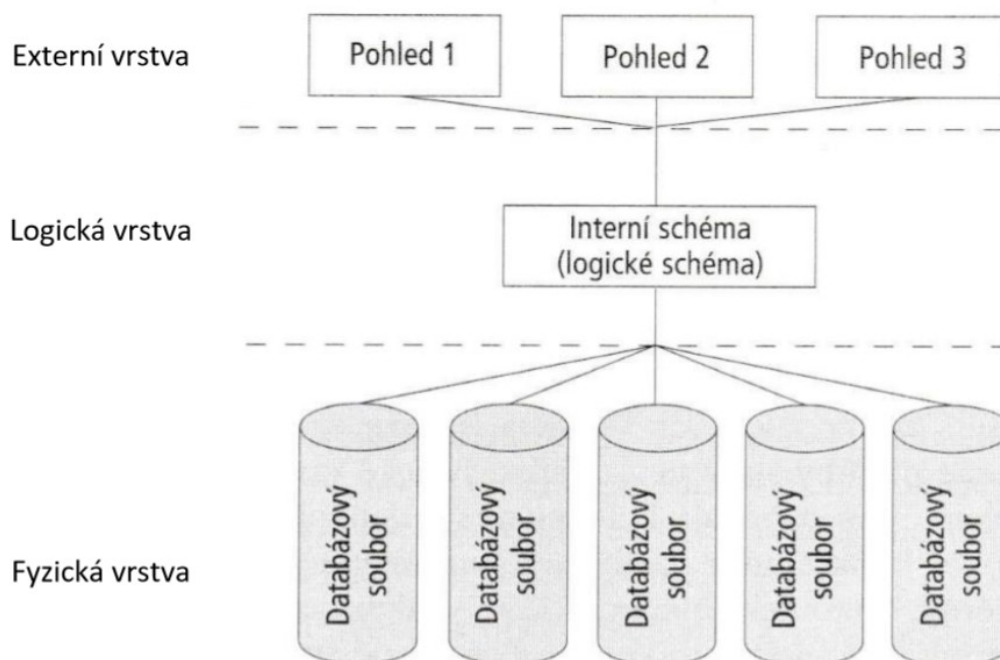
- přesouvání dat do fyzických datových souborů anebo opačně,
- správa současného přístupu více uživatelů k datům, včetně provádění opatření, která zabrání vzájemným konfliktům aktualizací provedených různými uživateli,
- správa transakcí, které znamenají současné vytvoření několika změn v databázi v rámci jedné nedělitelné jednotky,
- podpora dotazovacího jazyka, který tvoří množina příkazů pro načítání dat z databáze,
- mechanismy pro zálohování databáze a pro obnovu po havárii,
- bezpečnostní mechanismy, které zabraňují nepovolenému přístupu k datům a modifikacím. (Oppel, 2006)

## 2.2 Vrstvy datové abstrakce

Databáze nám nabízí výhodnou vlastnost, pomocí které umožňuje uživatelům u stejných jednou uložených dat, různé samostatné pohledy na tato data. Těmto pohledům se říká „uživatelské pohledy“. Zde můžeme považovat uživatele

za libovolnou osobu nebo aplikaci, která má možnost se přihlásit do databáze, a to za účelem uložení nebo načtení dat. Jeden z těchto pohledů je takový, že data uložíme do tabulkové aplikace. Poté musí všichni uživatelé pracovat s jedním společným pohledem. Druhá možnost je, aby si každý z uživatelů uložil data v samostatné kopii, ale jakmile jeden uživatel provede nějakou změnu, tak ostatní uživatele tuto změnu neuvidí. (Oppel, 2006)

V databázovém systému ale můžeme každému uživateli umožnit jiný pohled na stejná data, přičemž tyto pohledy mohou být speciálně přizpůsobeny každému uživateli na míru, a to i za předpokladu, že pracují společně nad jednou uloženou kopií dat. To je možné díky takzvaným vrstvám abstrakce, které vidíme na **Obrázku 2-1**. (Oppel, 2006)



**Obrázek 2-1:** Vrstvy abstrakce, zdroj: Oppel (2006, s. 19).

Na **Obrázku 2-1** vidíme architekturu, která se rychle stala základem při vývoji a výzkumu databázových systémů. Na této architektuře je postavena

většina moderních databází, přičemž se skládá ze tří základních vrstev, a to z fyzické, logické a externí vrstvy. V původní architektuře byla ještě konceptuální vrstva, která je ze schématu vyřazena, a to z důvodu, protože jí žádný z moderních výrobců databází neimplementuje. (Oppel, 2006)

### **2.2.1 Fyzická vrstva**

Fyzická vrstva obsahuje datové soubory, do kterých probíhá ukládání veškerých dat v příslušné databázi. V konkrétním databázovém systému může být jedna databáze uložena v několika souborech, které jsou často umístěny na více fyzických diskových jednotkách. U tohoto rozdělení mohou všechny diskové jednotky pracovat souběžně a databáze tak dosáhne vyššího výkonu. Jednou z výjimek ale je aplikace Microsoft Access, kde je celá databáze uložena v jednom fyzickém datovém souboru. Toto rozdělení ovšem znamená, že se databázový systém nedokáže přizpůsobit k práci většího počtu pracujících uživatelů, proto není vhodné pro velké podnikové databázové systémy. (Oppel, 2006)

### **2.2.2 Logická vrstva**

Logická vrstva se také někdy označuje pojmem schéma, jenž označuje kolekci všech datových položek, které jsou uloženy v příslušné databázi. Podle databázového systému může být schéma tvořeno množinou dvojrozměrných tabulek. (Oppel, 2006)

### **2.2.3 Externí vrstva**

Externí vrstva neboli externí model je vrstva, ve které se k databázi připojují uživatelé a aplikační programy, které s touto vrstvou dále pracují a vytváří v ní dotazy. Databázový systém zajišťuje transformaci z vybraných položek datových struktur v logické vrstvě do konkrétního uživatelského pohledu. Jelikož se uživatelské pohledy vytvářejí v této externí vrstvě, tak jsou definovány a ukládány do databáze, ve které je může kdokoliv znovu využít. (Oppel, 2006)

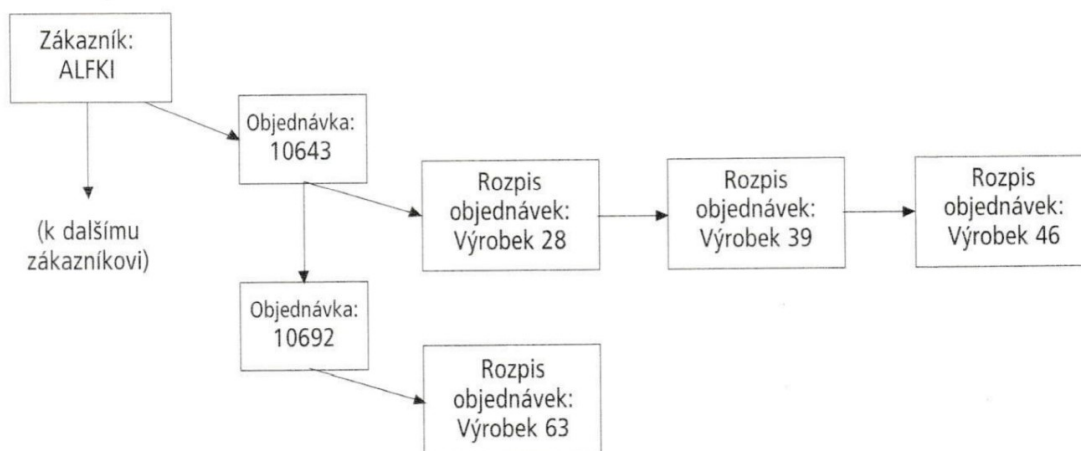
## **2.3 Databázový model**

Databázový model je vlastně architektura a pomocí této architektury databázový systém ukládá objekty do databáze, které poté vzájemně provazuje.

Níže jsou popsány nejrozšířenější modely databází, se kterými se můžeme setkat. (Oppel, 2006)

### 2.3.1 Hierarchický model

Nejranější databáze byly stavěny na hierarchickém modelu, který se vyvinul ze souborových systémů. Veškeré záznamy v tomto modelu jsou uspořádány do hierarchie, která je podobná například organizačnímu diagramu. Každý záznam je zde propojený pomocí ukazatelů, které obsahují adresu příslušného svázaného záznamu. Ukazatel tak vlastně počítači říká, kde se propojený záznam fyzicky nachází, toto se trochu podobá například poštovní adrese s ulicí a číslem domu nebo adrese URL. A toto znamená, že každý ukazatel definuje vztah nadřízený – podřízený záznam neboli rodič – potomek, tento vztah odznačujeme také jako relaci jedna k více. Díky této relaci může mít jeden rodič více potomků, ale každý potomek pouze jednoho rodiče. Jednou z nejrozšířenějších hierarchických databází býval systém IMS od firmy IBM. (Oppel, 2006)



**Obrázek 2-2:** Hierarchický model, zdroj: Oppel (2006, s. 26).

**Obrázek 2-2** zobrazuje hierarchický model, na kterém jsou vidět záznamy Zákazník, Objednávka a Rozpis objednávek.

V hierarchickém modelu je vytvořená hierarchie, ve které je nahoře umístěný rodič a směrem dolů nebo do strany je umístěný potomek. V tomto modelu poté vznikají tyto nevýhody:

- složité vkládání a rušení záznamů,
- v některých situacích nepřírozená organizace dat. (Farana, 1995)

### 2.3.2 Síťový model

Síťový model vznikl ve stejném období jako hierarchický model. Nejrozšířenější databází, která je postavena na síťovém modelu je systém IDMS, který vytvořila společnost Cullinane. Základem tohoto modelu jsou vzájemně propojené množiny záznamů. Celá databáze je složená ze dvou hlavních množin, které jsou množina záznamů a množina spojek. Ty jsou zvláštním typem záznamu o dvou položkách obsahující fyzické adresy záznamů. Síťový model je na rozdíl od hierarchického modelu podporuje vazby M:N. Ve stejné době vznikl relační model, který všechny ostatní modely vytlačil. (Oppel, 2006)



**Obrázek 2-3:** Síťový model, zdroj: Oppel (2006, s. 27).

### 2.3.3 Relační model

Oba předchozí modely jsou složité, ale také mají jednu společnou nevýhodu a tou je, že jsou velmi nepružné. To znamená, že zpracování dat je možné pouze při průchodu po přesně stanovené cestě. Například u jednorázových dotazů, jako je třeba vyhledávání, je nutné prohledat celou databázi. Proto v roce 1970 Dr. Codd vytvořil relační model databází. Tento model má jednoduchou strukturu, ve které jsou data organizována ve dvourozměrných tabulkách, které se skládají z řádků a sloupců. Namísto propojování příbuzných záznamů pomocí ukazatelů s fyzickou adresou, jako tomu bylo v hierarchickém a síťovém modelu, se zde do každé z tabulek zapíše společná datová položka. (Oppel, 2006)





**Obrázek 2-4:** *Relační model, zdroj: Oppel (2006, s. 29).*

**Obrázek 2-4** vyjadřuje relace typu jedna k více neboli 1:N, přičemž z relace na obrázku je zřejmé, že jeden zákazník je svázán s více objednávkami a toto znázornění se označuje jako diagram entit. (Oppel, 2006)

#### 2.3.4 Objektově orientovaný model

Objektově orientovaný model vznikl už v sedmdesátých letech minulého století, ale k významnému komerčnímu využití došlo až v devadesátých letech. Hlavními důvody pro tento rozmach byly tehdejší relační systémy RDBMS, se kterými se nedalo pracovat se složitějšími datovými typy, jako jsou obrázky, zvukové záznamy a videosoubory. Dále k rozmachu tohoto modelu dopomohlo rozšíření sítě Internet a služby WWW. (Oppel, 2006)

Pojmem objekt je chápáno logické seskupení příbuzných dat a programové logiky, které reprezentují většinou věci nebo osoby z reálného světa, jako je například zákazník, zaměstnanec. Každý objekt se poté skládá z datové položky neboli proměnné. Příkladem těchto datových položek může být například identifikátor zákazníka nebo jméno zaměstnance. Dalším důležitým pojmem je metoda, která pracuje nad určitým objektem, se kterým provádí určitou funkci, jako je například aktualizace adresy zákazníka. (Oppel, 2006)

Hlavním rozdílem mezi objektově orientovaným objektem a ostatními uvedenými modely je vlastnost, pomocí které k proměnným můžeme přistupovat jedině pomocí metod. Této vlastnosti se říká zapouzdření objektu. (Oppel, 2006)

### **2.3.5 Objektově relační model**

I přes výhody se nakonec objektově orientované databáze nedokázaly dostatečně prosadit a používají se pouze v malém procentu databází, ve kterých je potřeba zpracovávat složitá data. Výrobci relačních databází, ale dokázali zakomponovat výhody objektově orientovaného modelu. Kombinací obou modelů poté vznikl objektově relační model. Tento model využívají největší výrobci databází jako například firma Oracle a IBM. (Oppel, 2006)

### **2.3.6 NoSQL databáze**

Pojem NoSQL je také často označován jako „not only SQL“. Což si můžeme vysvětlit tak, že tyto databáze na rozdíl od relačních databází ve většině případů nepoužívají pouze dotazovací jazyk SQL. Příkladem použití jiných dotazovacích jazyků mohou být jazyky GQL a HQL. Oba tyto jazyky vycházejí z jazyka SQL a byly vytvořeny společností Google. (Günzl, 2012)

Nejčastější uplatnění pro NoSQL databáze je v oblastech big data, cloud computingu a sociálních sítí, kde je potřeba ukládat obrovské množství dat. Právě pro velké objemy dat jsou tyto databáze vhodné, jelikož jsou v porovnání s relačními databázemi ve čtení a zápisu dat mnohem lépe optimalizované. (Günzl, 2012; Sadalage, Fowler, 2013)

Mezi prvními společnostmi, které začali NoSQL databáze využívat, je společnost Google, která vytvořila svůj databázový systém BigTable. Ten je významný tím, protože není řádkově orientovaným systémem, ale sloupcově orientovaným. Další společností poté byl Amazon, který vytvořil databázový systém Dynamo. (Günzl, 2012)

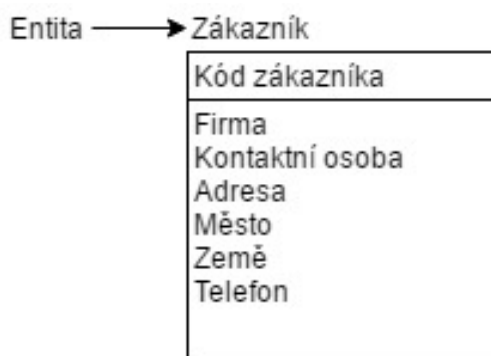
## **2.4 Datové modelování**

Tato podkapitola se zabývá pojmy jako jsou entity, atributy a relace, které jsou důležité pro základní pochopení a tvorbu tabulek v relačních databázích.

### **2.4.1 Entity**

Entita je množina objektů se shodnými vlastnostmi, o kterých shromažďujeme nějaká data. Nebo entitu můžeme popsat jednoduše tak, že jsou to předměty z reálného světa, které jsou pro nás zajímavé, proto je sledujeme a zaznamenáváme do databáze. Každou entitu identifikujeme jedinečným

jménem a seznamem vlastností, které se nazývají atributy. (Conolly, Begg a Holowczak, 2009; Oppel, 2006)



**Obrázek 2-5:** Entita, zdroj: Oppel (2006, s. 40).

Na **Obrázku 2-5** vidíme entitu zákazník, za její instanci bychom mohli například dosadit Adam Konečný. Každou entitu značíme obdélníkem. (Oppel, 2006)

#### 2.4.2 Atributy

Atribut je vlastnost, která entitu určitým způsobem charakterizuje nebo popisuje. Jinými slovy atributy představují to, co můžeme o entitách vědět. Každý atribut by měl být atomický, což znamená, že jeho hodnotu není možné dále smysluplně rozdělit do několika menších jednotek. Atributy se dále klasifikují na jednoduché a složené. (Conolly, Begg a Holowczak, 2009; Oppel, 2006)

Jednoduché atributy se skládají z jedné komponenty, a proto je nelze dále rozdělit. Složené atributy jsou složené z více komponent, které lze dále rozdělit na menší složky. (Conolly, Begg a Holowczak, 2009)



**Obrázek 2-6:** Atribut, zdroj: Oppel (2006, s. 40).

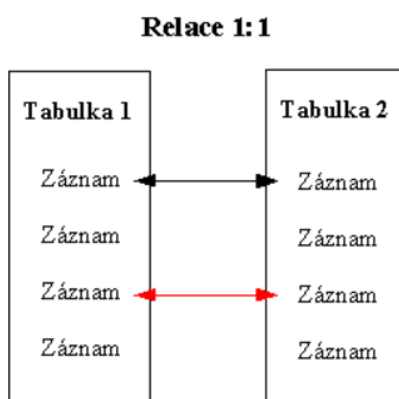
Na **Obrázku 2-6** vidíme atributy od entity zákazník. Atribut kód zákazníka je oddělen čarou od ostatních, protože je jednoznačný, takže každý zákazník musí mít svůj jedinečný kód zákazníka. Dále adresa je složený atribut, což znamená že jej můžeme snadno rozdělit do několika dalších údajů, například na atribut ulice a číslo popisné. (Oppel, 2006)

### 2.4.3 Relace

Relace je množina spojení mezi entitami. Jako u entit bychom měli každé spojení jedinečně identifikovat v rámci této množiny. Spojení, které je jedinečně identifikovatelné se nazývá výskyt relace. Každá relace se znázorňuje jako čára, která spojuje související entity. Bývají označeny jen v jednom směru, což znamená, že daná relace dává smysl pouze v jednom směru. Relace se dále rozlišují podle kardinality vztahu a to na 1:1, 1:N a M:N. (Conolly, Begg a Holowczak, 2009)

#### a) Relace 1:1

Relace typu 1:1 je vztah mezi entitami, kdy jednomu záznamu v první tabulce odpovídá maximálně jeden záznam v druhé tabulce. Tento typ relace není v praxi moc obvyklý, jelikož tyto entity je možné snadno sloučit do jednoho. I přesto se můžeme setkat s několika příklady, kdy se s tímto typem relace setkáme, například při rozdělení rozsáhlé tabulky, nebo při oddělení části tabulky z důvodu bezpečnosti. (Microsoft [online], 2017)

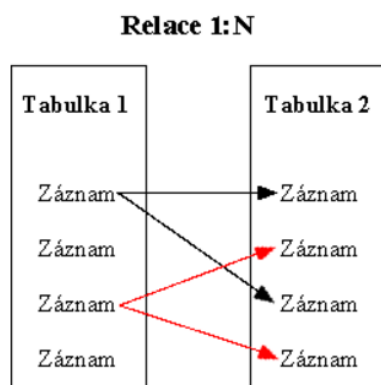


**Obrázek 2-7:** Relace 1:1, zdroj: Hauzar [Online] 2003.

#### b) Relace 1:N

Relace typu 1:N je vztah mezi entitami, kdy jednomu záznamu v první tabulce odpovídá více záznamů z druhé tabulky, ale naopak každý záznam druhé

tabulky odpovídá nejvýše k jednomu záznamu první tabulky. Tyto vztahy jsou v databázích velice časté, jelikož se jedná o základ relačních databází. (Microsoft [online], 2017)

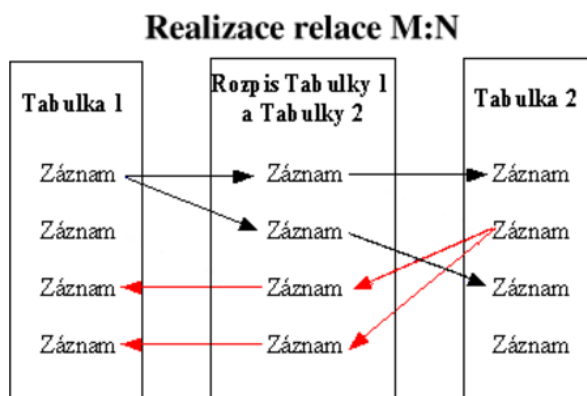


**Obrázek 2-8:** Relace 1:N, zdroj: Hauzar [Online] 2003.

Jednoduchý příklad relace 1:N je, když jeden zákazník může mít prodáno nula nebo mnoho objednávek a ke každé objednávce musí být přidělen právě jeden zákazník, kterému daná objednávka patří. Tento vztah můžeme vidět na **Obrázku 2-8**.

### c) Relace M:N

Relace typu M:N je zvláštním typem vztahu mezi dvěma entitami, u kterého může být záznam první tabulky spojen s nula, jedním nebo více záznamy, ale i opačně. Abychom mohli vyjádřit relaci M:N, tak musíme vytvořit třetí tabulku, která se většinou nazývá spojená tabulka, jež rozdělí relaci typu M:N na dvě relace 1:N. V praxi jsou relace tohoto typu velice běžné. (Microsoft [online], 2017; Oppel, 2006)



**Obrázek 2-9:** Relace M:N, zdroj: Hauzar [Online] 2003.

#### 2.4.4 Klíče relace

Každý záznam v tabulce musí být jedinečný. To znamená, že musíme být schopni určit sloupce nebo jejich kombinace, které tuto jedinečnost zajišťují. (Conolly, Begg a Holowczak, 2009)

##### a) Superklíč

Superklíč je sloupec nebo množina sloupců, které dokáží jedinečně identifikovat záznam v relaci. Superklíč může obsahovat sloupce, které pro jedinečné určení záznamu nejsou potřebné, proto je lepší se zaměřit na klíče, tuto podmínku splňují. (Conolly, Begg a Holowczak, 2009)

##### b) Kandidátní klíč

Kandidátní klíč je super klíč, který obsahuje minimální počet sloupců nutných k jedinečné identifikaci záznamů. (Conolly, Begg a Holowczak, 2009)

##### c) Primární klíč

Primární klíč je kandidátní klíč, který je vybrán, aby jedinečně určil záznamy v tabulce. (Conolly, Begg a Holowczak, 2009)



Obrázek 2-10: Primární klíč, zdroj: vlastní.

##### d) Cizí klíč

Cizí klíč je sloupec nebo skupina sloupců v jedné tabulce, která je shodná s kandidátním klíčem některé tabulky. (Conolly, Begg a Holowczak, 2009)

## 2.5 Normalizace

Normalizace je technika, která slouží k vytvoření sady tabulek s minimální redundancí. V roce 1972 E. F. Codd vyvinul techniku normalizace, která slouží pro podporu databází založených na relačním modelu. V rámci této techniky se často provádí řada testů v tabulce, aby se určilo, zda jsou dodržena pravidla

pro normální formu. Existuje několik těchto forem, nejužívanější z nich je první normální forma (1NF), dále druhá normální forma (2NF) a třetí normální forma (3NF). (Conolly, Begg a Holowczak, 2009)

První normální forma (1NF) je jedinou normální formou, která je důležitá pro vytvoření vhodných tabulek pro relační databázi. Všechny ostatní formy jsou volitelné, ale z důvodu abychom předcházeli anomáliím, se doporučuje používat třetí normální forma (3NF). (Conolly, Begg a Holowczak, 2009)

### 2.5.1 První normální forma

První normální formě (1NF) odpovídá tabulka, která neobsahuje žádné atributy s násobnými hodnotami. Jinými slovy, žádná buňka v této tabulce by neměla obsahovat více hodnot, nejde dále rozdělit a má svůj primární klíč. (Oppel, 2006)

Zákazníci		
ID	Jméno	Telefonní číslo
D001	Jiří Lingner	553 682 564, 603 926 865, 554 678 789
D002	Kamil Novák	701 878 567, 558 757 878
D003	Martina Bendová	604 688 890

**Tabulka 2-1:** Nenormalizovaná, zdroj: Conolly, Begg a Holowczak, 2009 (Upraveno).

Zákazníci			Telefonní čísla	
ID	Jméno	Příjmení	ID	Telefonní číslo
D001	Jiří	Lingner	D001	553 682 564
D002	Kamil	Novák	D001	603 926 865
D003	Martina	Bendová	D001	554 678 789
			D002	701 878 567
			D002	558 757 878
			D003	604 688 890

**Tabulka 2-2:** 1NF, zdroj: Conolly, Begg a Holowczak, 2009 (Upraveno).

**Tabulka 2-1** nesplňuje první normální formu (1NF). Jak lze vidět, tabulka obsahuje sloupec *Telefonní číslo*, ve kterém se nachází více telefonních čísel u jednoho záznamu. Z tohoto důvodu by bylo vhodné tato telefonní čísla rozdělit tak, aby pod jedním řádkem tabulky bylo pouze jedno *Telefonní číslo*. Následně je nutné rozdělit *Jméno* a *Příjmení* do jednotlivých sloupců, jak to ukazuje **Tabulka 2-2**.

### 2.5.2 Druhá normální forma

Tabulka je v druhé normální formě (2NF), pokud každý neklíčový atribut plně závisí na primárním klíči, a to na celém klíči, nejen na nějaké jeho podmnožině. Aby tato podmínka byla splněna, musí se primární klíč skládat z více než jednoho atributu. Pokud tedy primární klíč zahrnuje jen jedno pole, tabulka splňuje druhou normální formu (2NF), pokud také splňuje první normální formu (1NF). (Conolly, Begg a Holowczak, 2009)

Auta zákazníků				
Kod Auta	Značka	ID	Příjmení	Stav
165-01	Audi	D001	Lingner	Opraveno
165-02	BMW	D002	Novák	Neopraveno
165-03	Škoda	D003	Bendová	Neopraveno
165-04	Opel	D004	Horvátová	Opraveno
165-05	Škoda	D005	Čumpalík	Neopraveno
165-06	Opel	D006	Václavík	Opraveno

**Tabulka 2-3:** Nesplňující 2NF, zdroj: Conolly, Begg a Holowczak, 2009 (Upraveno).

Zákazníci		Auta		Stav		
ID	Příjmení	Kód Auta	Značka	Kód Auta	ID	Stav
D001	Lingner	165-01	Audi	165-01	D001	Opraveno
D002	Novák	165-02	BMW	165-02	D002	Neopraveno
D003	Bendová	165-03	Škoda	165-03	D003	Neopraveno
D004	Horvátová	165-04	Opel	165-04	D004	Opraveno
D005	Čumpalík	165-05	Škoda	165-05	D005	Neopraveno
D006	Václavík	165-06	Opel	165-06	D006	Opraveno

**Tabulka 2-4:** 2NF, zdroj: Conolly, Begg a Holowczak, 2009 (Upraveno).

**Tabulka 2-3** nesplňuje druhou normální formu (2NF), jelikož pole *Příjmení* závisí jen na jedné části primárního klíče a to na *ID*. Z tohoto důvodu mohou data trpět anomáliemi aktualizace, jelikož při odstranění auta může dojít ke ztrátě informací o zákazníkovi. Řešením této anomálie je rozdělení tabulky na více samostatných tabulek, které lze vidět v **Tabulce 2-4**.

### 2.5.3 Třetí normální forma

Třetí normální forma (3NF) se týká tabulek, které plní formu (1NF) a (2NF), dále ke splnění (3NF) je potřeba odstranit a oddělit veškerá data, která nejsou v přímém vztahu s danou tabulkou. To znamená, že hodnoty ve sloupcích



tabulky, které nepatří k primárnímu klíči, jsou determinovány pouze se sloupci primárního klíče. (Conolly, Begg a Holowczak, 2009)

#### 2.5.4 Boyce-Coddova normální forma

Boyce-Coddova normální forma je vylepšená třetí normální forma. (BCNF) říká, že mezi kandidátními klíči nesmí být žádná funkční závislost. (Oppel, 2006)

Na relaci v (BCNF) jsou kladeny tyto požadavky:

- tabulka musí být ve třetí normální formě,
- v tabulce nesmí být žádný určující atribut, který by byl primárním nebo kandidátním klíčem. (Oppel, 2006)

#### 2.5.5 Čtvrtá a pátá normální forma

Dále také existuje čtvrtá a pátá normální forma, ale tyto formy už nejsou v praktickém návrhu databází moc často využívány. (Microsoft [online], 2017)

Čtvrtá normální forma (4NF) se zabývá vztahy uvnitř primárního klíče. K splnění (4NF) je potřeba, aby tabulka splňovala (BCNF) a také popisovala pouze souvislost mezi primárním klíčem a atributy. (Oppel, 2006)

Pátá normální forma (5NF) vzniká u tabulek, které splňují (4NF) a nelze je dále rozložit bez toho, aniž by došlo ke ztrátě informací. (Oppel, 2006)

## 2.6 Transakce

Transakce je konečná uspořádaná posloupnost operací, které musí být všechny úspěšně provedeny, nebo nejsou provedeny, pokud dojde při provádění k chybě. Nejdůležitější vlastnosti transakcí označuje anglická zkratka ACID. (Oppel, 2006)

- **Atomičnost (Atomicity)** – transakce se musí celá úspěšně dokončit, v tom případě musí zachovat veškeré změny. Pokud ovšem selže, musí se všechny změny vrátit zpět,
- **Konzistentnost (Consistency)** – transakce převádí daný systém z jednoho konzistentního stavu do druhého,

- **Izolace (Isolation)** – operace dané transakce jsou izolovány od operací jiných transakcí,
- **Trvanlivost (Durability)** – po úspěšném provedení transakce jsou výsledky perzistentně uloženy na paměťové médium. (Oppel, 2006)

## 2.7 Microsoft Access

Microsoft Access je desktopová relační databázová aplikace, vytvořená společností Microsoft. Je součástí sady programů Microsoft Office. Využívá se k vytváření a správě relačních databázových aplikací. (Microsoft [online], 2017)

Aplikace Access představuje pro malé podniky a vývojáře, kteří pro ně vytvářejí aplikace, veškeré vývojářské nástroje. Vše, co je potřeba k uchovávání a správě dat, které slouží k provozu podniku. Pokud propojíme aplikaci Access s Microsoft SQL Serverem, tak dostaneme ideální nástroj pro vytvoření aplikací středně velkých podniků. (Viescas, 2008)

### 2.7.1 Datové typy

Datový typ si můžeme představit jako sadu vlastností, které platí pro všechny hodnoty obsažené v poli. V aplikaci Microsoft Access 2016 se datové typy dělí na 12 různých typů:

- **Krátký text** – alfanumerická data o velikosti až 255 znaků,
- **Dlouhý text** – velké množství alfanumerických dat o velikosti až 64 000 znaků,
- **Číslo** – pole k uložení číselných dat,
- **Datum a čas** – pole do kterého lze uložit datумы a časové hodnoty,
- **Měna** – peněžní částky uložené s přesností na 4 desetinná místa,
- **Automatické číslo** – jedinečná číselná hodnota, která je automaticky generovaná,
- **Ano/Ne** – datový typ Boolean, ukládá se jako numerická hodnota, kdy (0) označuje stav nepravda a (1) pravda,
- **Objekt OLE** – do pole lze uložit obrázky, grafy nebo jiné objekty z Windows do velikosti až 2 GB,

- **Hypertextový odkaz** – odkaz na dokument v rámci internetu, intranetu nebo místního počítače,
- **Příloha** – do pole lze přikládat soubory,
- **Počítané** – v poli můžeme vytvořit výraz, který využívá data z jiných polí. (Microsoft [online], 2017)

### 2.7.2 Tabulky

Tabulky jsou nejdůležitější částí databáze, které slouží k uchovávání údajů, s kterými se dále pracuje v dalších objektech aplikace Access. Tyto objekty si popíšeme později. Při navrhování databáze je důležité si správně navrhnout tabulky a dobře je spojit pomocí relací. Dále je potřeba v tabulkách zvolit vhodné datové typy, ty si zvolíme podle druhu dat, které budou uloženy v jednotlivých polích. (Belko, 2014)

### 2.7.3 Dotazy

Dotazy slouží k zobrazení údajů z jedné nebo více tabulek, případně jiných dotazů z databáze. Pomocí dotazů lze vybrat k zobrazení jen některá pole, také je uspořádat v jiném pořadí, než jsou v tabulce a přidat různá kritéria na zobrazení údajů. Kromě kritérií lze v rámci dotazu vytvářet i výpočty. Další prospěšnou funkcí v rámci dotazů jsou souhrnné dotazy, které dokáží například spočítat sumu. Kromě souhrnných dotazů existují i další typy, jako jsou například akční, křížový, sjednocovací, předávající a definiční. (Belko, 2014)

### 2.7.4 Formuláře

Formuláře slouží k práci s údaji a vytvářejí uživatelské rozhraní databázové aplikace. Jsou založené na údajích z tabulek a dotazů. S formuláři přicházejí do styku koneční uživatelé, kteří pracují s daty, měly by jim také zefektivnit a urychlit práci s databází. Pokud jsou formuláře správně navrženy, tak mohou také zabránit k zadání nesprávných dat. (Belko, 2014)

Formuláře hlavně využívají k zobrazování dat ve srozumitelné formě skryté dotazy. Jejich funkcionalitu lze poté značně vylepšit pomocí příkazů v jazyce VBA. Můžeme je také použít pouze pro vstup nových dat do databáze nebo z nich můžeme tisknout informace, ale většinou k tisku spíše používáme sestavy. (Shepherd, 2012)

### **2.7.5 Sestavy**

Sestavy nejčastěji slouží k vytvoření tiskových výstupu údajů z tabulek nebo dotazů. Jsou to objekty databáze, které se využívají k prezentaci informací z dané databáze pro některý z následujících účelů:

- zobrazení nebo distribuce souhrnu dat,
- archivní snímky dat,
- poskytování údajů o jednotlivých záznamech,
- vytváření popisků. (Microsoft [online], 2017)

### **2.7.6 Makra**

Makro je nástroj, kterým lze automatizovat úkoly a přidávat funkce do formulářů, sestav a ovládacích prvků. V aplikaci Access si můžeme představit makra jako značně zjednodušený programovací jazyk. Makra můžou přidávat funkce bez psaní kódu v modulu Visual Basic for Applications (VBA), díky tomu je použití maker pro uživatele mnohem jednodušší, než vytvářet stejné příkazy v jazyce (VBA). (Microsoft [online], 2017)

### **2.7.7 Moduly**

Moduly se používají na vytváření programového kódu v jazyce Visual Basic for Applications (VBA), který slouží podobně jako makra k automatizaci úloh. Zde už je ale potřeba ovládat tento programovací jazyk. (Belko, 2014)

VBA je programovací jazyk vyvinutý společností Microsoft, který se využívá v aplikacích Microsoft Office. Byl vydán v roce 1993 a aktuálně existuje ve verzi 7.1 od verze Office 2013. (Viescas, 2008)

VBA se využívá proto, jelikož některé úkony pomocí maker jsou velmi omezené. Například potřebujeme samostatné zpracování chyb v aplikaci a potřebuje vytvořit novou funkci, kterou makra nepodporují. (Viescas, 2008)

## 2.8 Zákony a vyhlášky pro sběrné suroviny

V této podkapitole jsou popsány veškeré zákony a vyhlášky, kterými se musí řídit sběrné suroviny na území ČR.

### 2.8.1 Katalog odpadů

V této podkapitole práce je popsán katalog odpadů, který využívají všechny sběrné suroviny provádějící svou činnost na území České republiky, a to podle vyhlášky 93/2016 Sb.

Katalog odpadů dělí jednotlivé druhy odpadu podle katalogových čísel, které se zapisují v šestimístních kódech, ve kterých první dvojčíslí označuje jednu z dvaceti skupin odpadu, druhé dvojčíslí podskupinu odpadu a třetí dvojčíslí druh odpadu. Jestliže se pro daný odpad nenajde odpovídající katalogové číslo ve skupinách 01 až 12 a 17 až 20, tak se hledá katalogové číslo ve skupinách 13, 14 a 15. Pokud se katalogové číslo nenajde ani v těchto třech skupinách, tak se hledá ve skupině 16. Nakonec jestli se nenajde vhodné katalogové číslo ani v této skupině, tak se odpadu přidělí katalogové číslo končící dvojčíslím 99. (Ministerstvo životního prostředí [online], 2017)

Pokud se jedná o jakýkoli nebezpečný odpad, tak se značí u čísla v katalogu odpadů symbolem „\*“ . (Ministerstvo životního prostředí [online], 2017)

01	Odpady z geologického průzkumu, těžby, úpravy a dalšího fyzikálního a chemického zpracování nerostů a kamene
02	Odpady z prvovýroby v zemědělství, zahradnictví, myslivosti, rybářství, lesnictví a z výroby a zpracování potravin
03	Odpady ze zpracování dřeva a výroby desek, nábytku, celulózy, papíru a lepenky
04	Odpady z kožedělného, kožešnického a textilního průmyslu
05	Odpady ze zpracování ropy, čištění zemního plynu a z pyrolytického zpracování uhlí
06	Odpady z anorganických chemických procesů
07	Odpady z organických chemických procesů
08	Odpady z výroby, zpracování, distribuce a používání nátěrových hmot (barev, laků a smaltů), lepidel, těsnicích materiálů a tiskařských barev

09	Odpady z fotografického průmyslu
10	Odpady z tepelných procesů
11	Odpady z chemických povrchových úprav, z povrchových úprav kovů a jiných materiálů a z hydrometalurgie neželezných kovů
12	Odpady z tváření a z fyzikální a mechanické úpravy povrchu kovů a plastů
13	Odpady olejů a odpady kapalných paliv (kromě jedlých olejů a odpadů uvedených ve skupinách 05 a 12)
14	Odpady organických rozpouštědel, chladiv a hnacích médií (kromě odpadů uvedených ve skupinách 07 a 08)
15	Odpadní obaly, absorpční činidla, čisticí tkaniny, filtrační materiály a ochranné oděvy jinak neurčené
16	Odpady v tomto katalogu jinak neurčené
17	Stavební a demoliční odpady (včetně vytěžené zeminy z kontaminovaných míst)
18	Odpady ze zdravotní nebo veterinární péče a /nebo z výzkumu s nimi souvisejícího (s výjimkou kuchyňských odpadů a odpadů ze stravovacích zařízení, které bezprostředně nesouvisejí se zdravotní péčí)
19	Odpady ze zařízení na zpracování (využívání a odstraňování) odpadu, z čistíren odpadních vod pro čištění těchto vod mimo místo jejich vzniku a z výroby vody pro spotřebu lidí a vody pro průmyslové účely
20	Komunální odpady (odpady z domácností a podobné živnostenské, průmyslové odpady a odpady z úřadů) včetně složek z odděleného sběru

**Tabulka 2-5:** Skupiny katalogových čísel, zdroj: (Ministerstvo životního prostředí, 2017).

### 2.8.2 Zákon o odpadech

V této podkapitole jsou popsána pravidla, kterými se řídí veškeré firmy se zaměřením na sběr surovin provádějící svou činnost na území České republiky, a to podle zákona o odpadech 185/2001 Sb. V tomto zákoně jsou definovány tři důležité body, které bylo nutné implementovat do dané databázové aplikace při její tvorbě.

Prvním z bodů je povinnost provozovatele zařízení ke sběru nebo výkupu odpadu identifikovat osoby, od kterých má v úmyslu tento odpad odebrat nebo vykoupit, spolu s identifikováním odpadu a zařazení do správné skupiny v katalogu odpadu. Dále je nutné při ukládání do evidence zavést i datum odebrání nebo vykoupení tohoto odpadu, aby provozovatel sběrných surovin byl schopen tuto povinnost plnit, je oprávněn po fyzických osobách požadovat nahlédnutí do jejich občanských průkazů. V případě právnické firmy je nutné zjištění názvu obchodní firmy a adresy jejího sídla. (Podnikatel.cz [online], 2017)

Takto získané identifikační údaje je provozovatel povinen evidovat a uchovat po dobu pěti let od odebrání nebo vykoupení daného odpadu. Jestliže nedojde k ověření těchto údajů, tak provozovatel nesmí odpad odebrat nebo vykoupit. (Podnikatel.cz [online], 2017)

Druhý bod určuje povinnost provozovateli sběrných surovin poskytovat úplatu za odevzdané železo nebo barevné kovy pouze převodem peněžních prostředků. To pomocí poskytovatele platebních služeb nebo poštovních služeb formou poštovního poukazu. (Podnikatel.cz [online], 2017)

Třetím bodem je ohlašovací povinnost, která vzniká provozovatelům sběrných surovin. Povinnost vzniká pouze, pokud dojde k překročení limitu produkce či nakládání s odpady a to 100 kilogramů nebezpečného nebo 100 tun ostatního odpadu za rok. V tomto případě je provozovatel povinen podat Roční hlášení o produkci a nakládání s odpady dle přílohy 20 vyhlášky č. 383/2001 Sb. s termínem do 15. února za každou samostatnou provozovnu. V tomto hlášení je provozovatel povinen uvést jednotlivé dodavatele a odběratele za každou transakci, která byla uskutečněna. (Inisoft.cz [online], 2017)

### **3 Současná situace ve firmě**

V této části práce je souhrnné seznámení s podnikem, ve kterém je bakalářská práce tvořena. Poté se zde zabývám analýzou současné situace a také požadavky, které si firma kladla pro svou databázovou aplikaci.

#### **3.1 Charakteristika firmy**

Podnik, pro který je práce realizována, se zabývá výkupem železa, barevných kovů, kovového odpadu a papíru. Poté následným zpracováním a prodejem většinou velkým společnostem jako ArcelorMittal Ostrava a.s. a TŘINECKÉ ŽELEZÁRNY, a.s.. Firma Martin Trupar Upon vznikla v roce 1997, v té době se zabývala pouze výkupem barevných kovů a kovového odpadu. Od roku 2000 se poté rozšířil sortiment služeb o výkup papíru.

Firma je v obchodním rejstříku vedena jako živnost koncesovaná. Databázová aplikace je vyvíjena přímo pro konkrétní potřeby této živnosti, po otestování dostatečné funkčnosti a odladění chyb je ze strany podniku velký zájem o využívání této aplikace v pracovním nasazení.

#### **3.2 Analýza současné situace**

V současné době ve firmě neexistuje žádná databáze v elektronické podobě, vše je zapisováno ručně do pokladních knih a skladových karet. Tyto dokumenty se musí skladovat ve fyzické podobě ve skladu, ale časem začal být problém se skladováním, protože množství dokumentů se stále zvyšovalo. Tento systém se ve firmě využíval po celou dobu její existence a fungoval bez závažnějších potíží.

S postupem času, rostoucím množstvím zakázek a zvyšující se snahou o optimalizaci správy jednotlivých záznamů, vzniká potřeba vést veškeré záznamy v elektronické podobě. Proto se živnostník rozhodl pro vytvoření databázové aplikace, která by veškeré tyto činnosti sjednotila do jediné ucelené aplikace, která bude mnohem přehlednější, než současný systém zapisování dat.



### **3.3 Požadavky na databázovou aplikaci**

Veškeré požadavky, které si firma stanovila, byly projednány s daným živnostníkem na několika schůzkách, na kterých došlo k prodiskutování a upřesnění stanovených podmínek.

Hlavním požadavkem provozovatele bylo, aby databázová aplikace obsahovala informace o všech dodavatelích a odběratelích, se kterými se bude moci poté dále pracovat.

Dále mezi hlavní požadavky patří záznam všechny skladů firmy, ve kterých se odpady a vybavení skladují. Poté taky záznam veškerých druhů odpadu, aby podnik přesně věděl množství odpadu, které má na skladech.

Mezi další důležitý požadavek patří záznam veškerých příjmu a výdajů odpadů, společně se záznamem, jaké společnosti se pohyb odpadů týká. Důležitá je také možnost tyto záznamy dále upravovat.

Dalším požadavkem je, aby potřebné informace, týkající se dodavatelů, odběratelů a pohybu odpadů, bylo možné buď vytisknout, nebo uložit ve formátu PDF.

Posledním požadavkem pro databázovou aplikaci bylo jednoduché rozhraní, a hlavně intuitivní ovládaní, neboť budoucí uživatelé disponují pouze základními dovednostmi ve správě databáze firmy.

## 4 Návrh a implementace databázové aplikace

Kapitola se zabývá návrhem a následnou implementací databázové aplikace. První část kapitoly je věnována návrhu databáze, ve kterém je popsána tvorba tabulek a vztahy mezi nimi. Tyto vztahy jsou podrobně popsány společně s definováním jednotlivých primárních a cizích klíčů.

V druhé části kapitoly se nachází rozbor veškerých tabulek, které jsou navrženy pomocí programu Microsoft Access. V popisu tabulky je definováno, k čemu daná tabulka slouží, s následným odůvodněním, proč dané pole tabulky bylo použito. V rozboru jsou definovány jednotlivé primární a cizí klíče, které jsou v tabulkách použity. Nejdůležitější je popis specifikací polí, které jsou asociovány s tabulkami, jako je například jejich datový typ, velikost pole, vstupní maska, nutnost zadat dané pole a indexace pole.

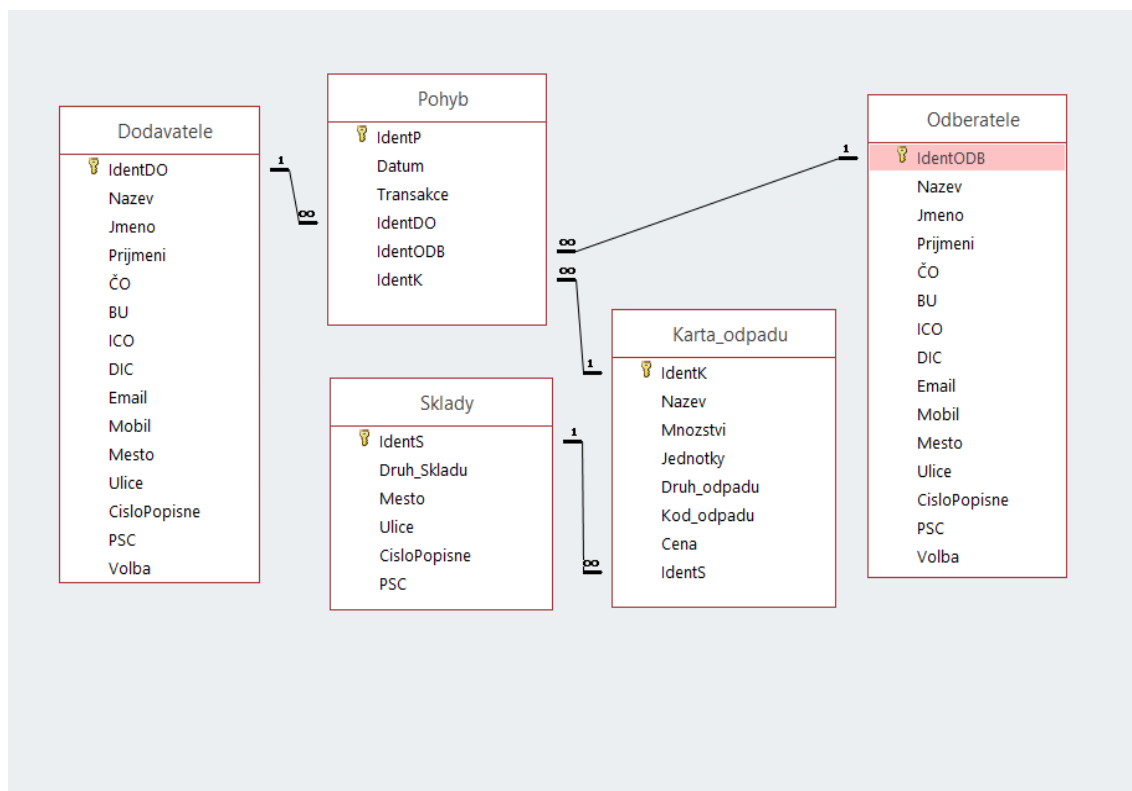
Třetí část se zabývá popisem jednotlivých formulářů, které jsou automatizovaný pro danou databázovou aplikaci pomocí programovacího jazyka Visual Basic for Applications nebo funkcí maker. Všechny formuláře jsou podrobně rozebrány jak z grafické stránky, tak i ze stránky funkční. Dále jsou zahrnuty další formuláře, které jsou vytvořeny pro přidávání nebo úpravu dat. Pro jednoduchý pohyb mezi jednotlivými formuláři je vytvořen navigační pás, který byl vytvořen pomocí speciálního navigačního formuláře.

V poslední části kapitoly bude databázová aplikace převedena do uživatelského prostředí, aby uživatel neměl přístup k jednotlivým objektům databáze. Z důvodu, aby neohrozil základní funkcionalitu databázové aplikace. Finální aplikace bude uložena na přenositelné médium (např. CD nebo flash disk).

## 4.1 Návrh databáze

Fyzický návrh databáze je tvořen pěti tabulkami, které jsou detailně popsány v následující podkapitole.

Na **Obrázku 4-1** lze vidět veškeré relační vztahy mezi těmito tabulkami v databázové aplikaci. Tyto vztahy jsou realizovány pomocí primárních a cizích klíčů.



**Obrázek 4-1:** Návrh databáze, zdroj: vlastní.

Na výše uvedeném **Obrázku 4-1** lze vidět tabulku *Sklady*, která je propojená pomocí primárního klíče *IdentS* s tabulkou *Karta\_odpadu*, ve které se nachází cizí klíč *IdentS*. Tabulka *Karta\_odpadu* je propojena pomocí primárního klíče *IdentK* s tabulkou *Pohyb*, kde se nachází cizí klíč *IdentK*. Následující tabulka *Pohyb* je propojena pomocí cizích klíčů *IdentDO* a *IdentODB* s tabulkami *Dodavatele* a *Odberatele*, ve kterých se nacházejí primární klíče *IdentDO* a *IdentODB*. Všechny tyto vztahy jsou realizovány pomocí relace 1:N.

## 4.2 Rozbor použitých tabulek

Podkapitola 4.2 popisuje praktickou realizaci jednotlivých tabulek, které jsou v databázové aplikaci použity. Ke každé z nich je přidán jednoduchý popis, který vysvětluje, k čemu tabulka slouží. U každého popisu se nachází přehledná tabulka, ve které jsou popsány jednotlivé položky spolu s datovým typem a vstupní maskou.

### 4.2.1 Tabulka Dodavatele

Tabulka Dodavatele v databázové aplikaci slouží k uložení jednotlivých dodavatelů, kteří pro sběru druhotných surovin dodávají železo, barevné kovy nebo papír.

Tabulka obsahuje patnáct polí, kde první pole je *IdentDO*, do kterého se ukládá identifikátor dodavatele. Toto pole je primárním klíčem, proto musí být jeho obsah jedinečný v celé databázi, aby bylo možné každého dodavatele jednoznačně identifikovat. Podobu identifikátoru má firma přesně stanovenou pomocí vstupní masky, kde první dva znaky zaujímají písmena a poslední tři číslice.

Druhé pole v tabulce, *Nazev*, zobrazuje název společnosti, která je dodavatelem, ale pouze v případě, že se jedná o právnickou osobu. Jestli že se jedná o fyzickou osobu, tak toto pole zůstane prázdné a využijí se další dvě pole *Jmeno* a *Prijmeni*. Do nich se vyplní jméno a příjmení daného dodavatele.

Následující pole *ČO* (číslo občanských průkazů) je nutné vyplnit v případě, že se jedná o fyzickou osobu, jelikož sběrné suroviny jsou povinny podle zákona č. 185/2001 Sb. odstavec (3) vést jednoznačnou identifikaci lidí, kteří donesou jakékoliv množství železa nebo barevných kovů. Firma zaznamenává čísla občanských průkazů.

Pole *BU* (běžný účet) je nutné vždy vyplnit z důvodu povinnosti vyplácet dodavatele, podle zákona č. 185/2001 Sb. odstavec (5) a (11), kteří jsou právnické či fyzické osoby, pouze v bezhotovostní formě. Údaj není povinný, pokud není uveden, vystavuje firma poštovní poukázku na adresu, uvedenou v poli *Mesto* (viz dále).

Další pole *ICO* (identifikační číslo osoby) a *DIC* (daňové identifikační číslo) jsou potřeba k uložení čísel, které slouží ke specifikování právnických osob.

Následující tři pole, pojmenovaná *Email*, *Mobil*, *Mesto*, slouží k uložení kontaktních informací o dodavateli.

Klíč	Název pole	Datový typ	Velikost pole	Vstupní maska	Je nutno zadat	Indexovat
PK	IdentDO	Krátký text	5	LL000	Ano	Ano (bez duplicity)
	Nazev	Krátký text	30	-	Ne	Ne
	Jmeno	Krátký text	30	-	Ne	Ne
	Prijmeni	Krátký text	30	-	Ne	Ne
	ČO	Krátký text	9	0000000 00	Ne	Ne
	BU	Krátký text	20	000000\_ 0000000 000V000 0	Ano	Ne
	ICO	Krátký text	8	0000000 0	Ne	Ne
	DIC	Krátký text	12	LL00000 00000	Ne	Ne
	Email	Krátký text	40	-	Ne	Ne
	Mobil	Krátký text	9	" +420 "000\ 000\ 000	Ne	Ne
	Mesto	Krátký text	30	-	Ano	Ne
	Ulice	Krátký text	30	-	Ano	Ne
	CisloPopisne	Krátký text	10	-	Ano	Ne
	PSC	Krátký text	5	000\ 00	Ano	Ne
	Volba	Číslo	Dlouhé celé č.	-	Ne	Ne

**Tabulka 4-1: Dodavatele, zdroj: vlastní.**

Poslední pole *Volba* je potřeba z důvodu zaznamenání, zda se jedná o dodavatele, který je fyzickou nebo právnickou osobou. Toto pole uživatel nevyplňuje, neboť je vyplněno automaticky aplikací v okamžiku, kdy uživatel vybere příslušnou právní formu dodavatele.

#### 4.2.2 Tabulka Odberatele

Tabulka *Odberatele* slouží v databázové aplikaci k uložení jednotlivých odběratelů, kterým sběrna druhotných surovin dodává železo, barevné kovy a papír. Odběratelé jsou ve většině případů právnické osoby.

Návrh tabulky *Odberatele* je téměř totožný jako tabulka *Dodavatele*. Jediná změna je v primárním klíči, který se zde nazývá *IdentODB*. V tomto poli je uložen identifikátor odběratele, který je deklarován vstupní maskou, ve které první dva znaky jsou zastoupeny písmeny a poslední tři číslicemi.

Klíč	Název pole	Datový typ	Velikost pole	Vstupní maska	Je nutno zadat	Indexovat
PK	IdentODB	Krátký text	5	LL000	Ano	Ano (bez duplicity)
	Nazev	Krátký text	30	-	Ne	Ne
	Jmeno	Krátký text	30	-	Ne	Ne
	Prijmeni	Krátký text	30	-	Ne	Ne
	ČO	Krátký text	9	000000000	Ne	Ne
	BU	Krátký text	20	000000\ - 00000000000\ /0000	Ano	Ne
	ICO	Krátký text	8	00000000	Ne	Ne
	DIC	Krátký text	12	LL00000000 00	Ne	Ne
	Email	Krátký text	40	-	Ne	Ne
	Mobil	Krátký text	9	" +420 "000\ 000\ 000	Ne	Ne
	Mesto	Krátký text	30	-	Ano	Ne
	Ulice	Krátký text	30	-	Ano	Ne
	CisloPopisne	Krátký text	10	-	Ano	Ne

	PSC	Krátký text	5	000\ 00	Ano	Ne
	Volba	Číslo	Dlouhé celé č.	-	Ne	Ne

**Tabulka 4-2: Odberatele, zdroj: vlastní.**

#### 4.2.3 Tabulka Sklady

Tabulka *Sklady* slouží v databázové aplikaci k uložení jednotlivých skladů, ve kterých je umístěno dané množství železa, barevných kovů nebo papíru. Sklady jsou děleny podle druhu odpadu, který je v nich uložen, nebo podle jiných materiálů. Například může nastat situace, kdy je potřeba ve skladě uložit pracovní nástroje a techniku, kterou využívá firma a potřebuje je uskladnit, nebo se sklad nachází v jiné provozovně.

Tabulka se skládá ze šesti polí, kde první pole je primární klíč, který se nazývá *IdentS*. Jeho formát je stejný, jako v případě ostatních tabulek, v nichž je omezena délka pole na pět znaků, kde první dva z nich jsou písmena a poslední tři číslice.

Další pole, které se nachází v tabulce, je *DruhSkladu*. Toto pole v sobě nese informaci, pro jaký účel daný sklad slouží, například jestli slouží k ukládání železa, barevných kovů, papíru nebo jako sklad pro nářadí a další materiál, který sběrna potřebuje uložit.

Klíč	Název pole	Datový typ	Velikost pole	Vstupní maska	Je nutno zadat	Indexovat
PK	IdentS	Krátký text	5	LL000	Ano	Ano (bez duplicity)
	DruhSkladu	Krátký text	30	-	Ano	Ne
	Mesto	Krátký text	30	-	Ne	Ne
	Ulice	Krátký text	30	-	Ne	Ne
	CisloPopisne	Krátký text	10	-	Ne	Ne
	PSC	Krátký text	5	000\ 00	Ne	Ne

**Tabulka 4-3: Sklady, zdroj: vlastní.**

Ostatní pole jako je *Mesto*, *Ulice*, *CísloPopisne* a *PSC* (poštovní směrovací číslo), slouží k uložení adresy daného skladu.

#### 4.2.4 Tabulka *Karta\_odpadu*

Tabulka *Karta\_odpadu* slouží v databázové aplikaci k uložení jednotlivého odpadu podle množství a druhu.

Tabulka se skládá z osmi polí, ve které první pole na nazváno *IdentK*. Pole je primárním klíčem, ve kterém má firma přesně definovanou jeho podobu pomocí vstupní masky.

Druhým polem v této tabulce je *Nazev*. Toto pole slouží k uložení názvu určitého odpadu tak, aby bylo snazší odpad identifikovat a rozeznat, zda se jedná o železo, barevné kovy nebo papír.

Následující pole *Mnozstvi* je určeno k zadání hmotnosti daného odpadu, datovým typem je číslo. V dalším poli se zadávají jednotky, ve kterých je odpad zvážen a uložen na sklad. Pole *DruhOdpadu* dále specifikuje, o jakou surovinu se jedná.

Následující pole *KodOdpadu* je nutné vyplnit podle vyhlášky 93/2016 Sb., ve které je definováno, do které skupiny odpadů daný materiál patří. Vyhláška přesně určuje, jaký formát má kód splňovat. Z toho důvodu je vytvořena vstupní maska, která obsahuje šest povinných číslic a jeden nepovinný znak (vyjádřen \*) v případě, že je jedná o nebezpečný odpad.

Poslední pole nazvané *IdentS*, slouží k identifikaci skladu, ve kterém je daný odpad uložen. Jedná se o stejné pole, které je uloženo v tabulce sklady, jenom s rozdílem, že zde vystupuje jako cizí klíč. Z tohoto důvodu je zde povolena duplicita, jelikož stejný identifikátor už obsahuje tabulka sklady.

Klíč	Název pole	Datový typ	Velikost pole	Vstupní maska	Je nutno zadat	Indexovat
PK	IdentK	Krátký text	5	LL000	Ano	Ano (bez duplicity)
	Nazev	Krátký text	30	-	Ne	Ne
	Mnozstvi	Číslo	-	-	Ano	Ne



	Jednotky	Krátký text	10	-	Ne	Ne
	DruhOdpadu	Krátký text	30	-	Ne	Ne
	KodOdpadu	Krátký text	7	000000C	Ano	Ne
	Cena	Měna	-	-	Ano	Ne
CK	IdentS	Krátký text	5	LL000	Ano	Ano (duplicita povolena)

**Tabulka 4-4:** *Karta\_odpadu*, zdroj: vlastní.

#### 4.2.5 Tabulka Pohyb transakcí

Tabulka *Pohyb transakcí* slouží v databázové aplikaci k uložení probíhajících transakcí, které jsou prováděny mezi dodavateli a odběrateli sběrných druhotných surovin.

Tato tabulka obsahuje šest polí, kde první je nazváno *IdentP*. Pole je primárním klíčem a slouží jako u všech ostatních tabulek k identifikaci dané transakce. Ostatní parametry jsou shodné s ostatními identifikátory v tabulkách.

Následující pole *Datum* slouží k uložení data, kdy daná transakce proběhla. Toto pole má nastavený datový typ datum a čas, který je pro tento typ dat nejvhodnější. Při vyplňování tabulky je nutné pole datum vyplnit.

Pole *Transakce* je určeno k výběru, zda daná transakce je příjmem nebo výdajem. Poslední pole *IdentDO*, *IdentODB* a *IdentK* jsou cizími klíči z tabulek *Dodavatel*, *Odberatel* a *Karty\_odpadu*. Jsou tvořeny stejnou vstupní maskou, ve které první dva znaky jsou zastoupeny písmeny a poslední tři číslicemi. Pomocí těchto polí lze snadno poznat, jakého dodavatele nebo odběratele daná transakce má a také jakého druhu odpadu se týká.

Klíč	Název pole	Datový typ	Velikost pole	Vstupní maska	Je nutno zadat	Indexovat
PK	IdentP	Krátký text	5	LL000	Ano	Ano (bez duplicity)
	Datum	Datum a čas	-	-	Ano	Ne
	Transakce	Číslo	10	-	Ne	Ne

CK	IdentDO	Krátký text	5	LL000	Ne	Ano (duplicita povolena)
CK	IdentODB	Krátký text	5	LL000	Ne	Ano (duplicita povolena)
CK	IdentK	Krátký text	5	LL000	Ano	Ano (duplicita povolena)

**Tabulka 4-5:** Pohyb transakcí, zdroj: vlastní.

### 4.3 Rozhraní databázové aplikace

V databázové aplikaci je pro pohyb mezi formuláři vytvořen jeden vodorovný navigační formulář, který slouží k pohybu mezi jednotlivými formuláři.



**Obrázek 4-2:** Navigační pás aplikace, zdroj: vlastní.

Navigační pás lze vidět na **Obrázku 4-2**. Grafický design databázové aplikace byl navržen podle požadavků firmy tak, aby byl snadný a intuitivní na ovládání, ale také dostatečně přehledný, aby se v aplikaci zorientovali i zaměstnanci, kteří s databázovými systémy nemají moc zkušeností. Barevné schéma pro aplikaci bylo vybráno tak, aby ladilo s logem firmy a bylo graficky jednotné.

#### 4.3.1 Formulář frmDomu

Po spuštění databázové aplikace se uživateli zobrazí formulář s názvem *frmDomu*. Tento formulář slouží jako úvodní strana pro databázovou aplikaci, pomocí které lze přistupovat k ostatním formulářům přes navigační pás v horní části domovského formuláře.

Skládá se ze tří částí, kde v první části (v záhlaví) se nachází datum a čas. Uprostřed je umístěno logo firmy společně s pozdravem, který se mění v závislosti na denní době, ve které se databáze spustí. V ranních hodinách se zobrazí pozdrav „dobré ráno“. Od 11:01 do 13:00 se zobrazí „dobré poledne“. Od 13:01 do 17:00 se vypíše ve formuláři „dobré odpoledne“ a od 17:01 do půlnoci „dobrý večer“. Poslední část formuláře (zápatí) obsahuje tlačítko Ukončit, které slouží k ukončení celé databáze.



**Obrázek 4-3:** *Formulář Domů, zdroj: vlastní.*

#### **4.3.2 Formulář frmDodavatele**

Tento formulář získáme po přechodu k položce dodavatelé na pásu karet navigačního formuláře. Slouží k prohlížení jednotlivých dodavatelů, jejich přidání, následné editaci a odstranění. K dispozici mají uživatelé také možnost zobrazit seznam všech dodavatelů, jejich následný tisk nebo mohou daný seznam dodavatelů uložit do souboru ve formátu PDF.

Konkrétní podoba formuláře je znázorněna na **Obrázku 4-4**, na kterém lze vidět jak grafickou, tak i funkční podobu formuláře dodavatelé. V horní části je vidět zápatí, ve kterém je umístěn název formuláře. Na pravé straně se nachází šest základních tlačítek, které slouží k práci s daty.

# Dodavatelé

Přidat

Seznam

Hledat podle Názvu

Upravit

Tisk

Odstranit

Uložit do PDF

Hledat

Zrušit

ID Dodavatele	DO090
Název	Mittal steel
Jméno	
Příjmení	
Číslo občanky	
Běžný účet	787779-8908967453/5645
IČO	67545607
DIČ	V86567678678
E-mail	info@mittal.com
Mobil	+420 878 987 897
Město	Ostrava
Ulice	Vitkovická
Číslo popisné	6778
PSČ	710 00

◀

◁

▷

▶

**Obrázek 4-4:** Formulář Dodavatelé, zdroj: vlastní.

Tlačítko přidat slouží k přidání nových dodavatelů do databáze. Následně je možné je upravit, pokud například došlo k nějaké změně informací o dodavateli. V případě, že již daného dodavatele nebude firma dále využívat, tak je možné jej odstranit. Při odstranění je uživatel dotázán, jestli chce skutečně položku odstranit, aby nedošlo k nechtěnému odstranění. Zde je nutné si dát pozor, zda tento dodavatel není zadán ve formuláři pohyb transakcí, protože by došlo k nekonzistenci dat a uživateli není povoleno tohoto dodavatele odstranit.

Následně vedle tlačítek vidíme vyhledávací pole, pomocí kterého mohou uživatelé vyhledat dodatele podle jeho názvu, jedná-li se o právnickou osobu nebo podle příjmení, jedná-li se o fyzickou osobu. Pokud uživatel bude chtít dané vyhledávání zrušit, tak k tomuto účelu se u vyhledávacího pole nachází tlačítko zrušit.

V zápatí formuláře se nacházejí navigační tlačítka, pomocí kterých mohou uživatelé přecházet mezi jednotlivými dodavateli, kteří jsou uloženi v databázi.

Přidání dodavatele	Upravení dodavatele
Právní forma <input type="radio"/> Fyzická osoba <input checked="" type="radio"/> Firma	Právní forma <input type="radio"/> Fyzická osoba <input checked="" type="radio"/> Firma
ID Dodavatele	ID Dodavatele DO090
Název	Název Mittal steel
Jméno	Jméno
Příjmení	Příjmení
Číslo občanky	Číslo občanky
Běžný účet	Běžný účet 787779-8908967453/5645
IČO	IČO 67545607
DIČ	DIČ VB6567678678
E-mail	E-mail info@mittal.com
Mobil	Mobil +420 878 987 897
Město	Město Ostrava
Ulice	Ulice Vítkovická
Číslo popisné	Číslo popisné 6778
PSČ	PSČ 710 00
OK Storno	OK Storno

**Obrázek 4-5:** Formuláře přidat a upravit dodavatele, zdroj: vlastní.

Při přidávání nového nebo úpravě stávajícího jsou otevřeny nové formuláře *frmPridatDodavatele* a *frmUpravitDodavatele*, které jsou znázorněny na **Obrázku 4-5**. Tyto formuláře obsahují celkem čtrnáct polí, ale uživatel vyplňuje nebo upravuje pouze jedenáct různých polí, které se mění v závislost na tom, zda je dodavatel fyzická osoba nebo firma. Do těchto polí poté uživatel vyplňuje nebo upravuje základní informace o dodavateli, které jsou pro firmu důležité.

#### 4.3.3 Formulář frmSklady

K tomuto formuláři se uživatel dostane po přechodu na položku sklady na pásu karet v horní části. Formulář *frmSklady* slouží k uložení jednotlivých skladů, ve kterých má firma uložen jakýkoliv odpad. Následně lze v tomto formuláři dále upravit, odstranit nebo vytvořit nový sklad. Také jako v předchozím formuláři *frmDodavatelé*, tak i zde je možné zobrazit seznam všech skladů, které lze vytisknout nebo uložit do souboru ve formátu PDF.

**Obrázek 4-6** zobrazuje podobu formuláře. Ten se skládá ze tří částí, kde v první části se nachází název formuláře spolu s šesti tlačítky a vyhledávacím polem.

ID Skladu	Druh skladu	Město	Ulice	Číslo popisné	PSČ
SK234	Měď	Opava	Myslbekova	2	746 01
SK239	Cín	Ostrava	Těšínská	567	710 00
SK555	Stroje	Opava	Myslbekova	2	746 01
SK567	Ocel	Opava	Myslbekova	2	746 01
SK598	Nerez	Ostrava	Vdovská	18	710 00
SK678	Papír	Ostrava	Těšínská	567	710 00
SK778	Nářadí	Opava	Myslbekova	2	746 01
SK879	Železo	Ostrava	Vdovská	18	710 00

**Obrázek 4-6:** Formulář sklady, zdroj: vlastní.

První tlačítko slouží k přidání skladů do databáze aplikace. Pomocí dalších tlačítek je možné jednotlivé položky upravit nebo odstranit. Při odstranění je nutné dbát na to, aby daný sklad nebyl uložen v tabulkách *Karta\_odpadu* nebo *Pohyb transakcí*, jelikož v takovém případě daný sklad nelze odstranit.

Vedle těchto tlačítek lze vidět vyhledávací pole, pomocí kterého můžou uživatelé vyhledat jednotlivý sklad podle parametru druh skladu. Tento parametr popisuje, jaký druh odpadu nebo materiálu je v daném skladu uložen. U tohoto pole jsou také umístěná dvě tlačítka, první tlačítko *Hledat* slouží k potvrzení vyhledání daného skladu. Druhé tlačítko *Zrušit*, pomocí kterého uživatel zruší dané vyhledávání a v náhledu zase uvidí všechny sklady, které se nachází v databázové aplikaci.

V zápatí formuláře se stejně jako v předchozím formuláři nacházejí tlačítka určená k navigaci mezi jednotlivými záznamy.

Přidání skladů	Upravení skladů
ID Skladu	SK234
Druh skladu	Měď
Město	Opava
Ulice	Myslbekova
Číslo popisné	2
PSČ	746 01
OK Storno	OK Storno

**Obrázek 4-7:** Formuláře Přidat a Upravit sklad, zdroj: vlastní.

Při použití tlačítek *Přidat* nebo *Upravit* se otevřou nové formuláře *frmPridatSklady* a *frmUpravitSklady*, podoba těchto formulářů je znázorněna na Obrázku 4-7. Skládají se celkem ze šesti polí, do kterých uživatel může ukládat nebo upravovat základní informace o skladu.

#### 4.3.4 Formulář frmKartaOdpadu

Jeden z hlavních formulářů *frmKartaOdpadu* uživatel získá po přechodu na navigačním pásu na položku *Odpad*. Tento formulář slouží k uložení jednotlivých položek odpadu podle jeho druhu. Uživatel v náhledu přehledně vidí, jaké množství určitého druhu odpadu mají na skladě společně s jeho kódem a cenou, za jakou byl nakoupen. Další funkce tohoto formuláře jsou zobrazení seznamu všech odpadů, který lze následně vytisknout nebo uložit jako soubor ve formátu PDF.

Karta odpadu							
		Přidat	Seznam	Hledat podle Odpadu			
		Upravit	Tisk				
		Odstranit	Uložit do PDF	Hledat	Zrušit		
ID Karty	Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena za jednotku	ID Skladu
KA234	Barevné kovy	3000	kg	Hliník	174402	30,00 Kč	SK555
KA340	Nebezpečný odpad	40	kg	Autobaterie	175092*	5,00 Kč	SK678
KA567	Papír	67	kg	Lepenka	150101	5,00 Kč	SK678
KA579	Barevné kovy	1000	kg	Mosaz	170401	50,00 Kč	SK239
KA769	Papír	2000	kg	Časopisy	150102	2,00 Kč	SK678
KA789	Železo	50	kg	Železný odpad	174405	20,00 Kč	SK879
KA907	Barevné kovy	70	kg	Měď	170401	200,00 Kč	SK234
		0				0,00 Kč	

**Obrázek 4-8:** Formulář Karta odpadu, zdroj: vlastní.

Na **Obrázku 4-8** lze vidět grafickou podobu formuláře, který je rozdělen na tři části. První část (záhlaví) je složena stejně jako předchozí formuláře z názvu formuláře, dále z šesti tlačítek a vyhledávacího pole. V druhé části je náhled na jednotlivé odpady, které jsou uloženy v databázi firmy. Poslední část (zápatí) se skládá z navigačních tlačítek, které uživatel může použít ke změně označení jednotlivých položek odpadů.

Hodnota pole *Cena za jednotku* je vedeno jako průměrná cena a vypočítáváno vždy při přidání pohybu na skladě.

Při použití tlačítek *Přidat* nebo *Upravit* se otevřou nové formuláře *frmPridatOdpad* a *frmUpravitOdpad*, podoba těchto formulářů je znázorněna na **Obrázku 4-9**. Skládají se celkem z osmi polí, do kterých uživatel může ukládat nebo upravovat důležitá data o odpadu.

Přidání odpadu	Upravení odpadů
ID Karty	<input type="text" value="KA234"/>
Název	<input type="text" value="Barevné kovy"/>
Množství	<input type="text" value="3000"/>
Jednotky	<input type="text" value="kg"/>
Druh odpadu	<input type="text" value="Hliník"/>
Kód odpadu	<input type="text" value="1744-02"/>
Cena	<input type="text" value="30,00 Kč"/>
ID Skladu	<input type="text" value="SK555"/>
OK Storno	OK Storno

**Obrázek 4-9:** Formuláře přidat a upravit odpad, zdroj: vlastní.

#### 4.3.5 Formulář *frmPohybTransakci*

Formulář *frmPohybTransakci* slouží ke sledování nebo zapisování jednotlivých transakcí, které byly firmou provedeny. Tento formulář je dostupný v navigačním pásu pod položkou *Pohyb transakcí*.

Jak lze vidět na **Obrázku 4-10**, formulář se skládá ze záhlaví, ve kterém se nachází název formuláře, dále tlačítka, které slouží pro přidání, odstranění a úpravě nové položky. Uživatel má možnost si zobrazit seznam jednotlivých transakcí, který je možné vytisknout nebo uložit ve formátu PDF. Vedle těchto tlačítek se zde nachází vyhledávací pole, pomocí kterého uživatel může vyhledat určitou transakci, to podle druhu transakce. Také zde můžeme zjistit, zda se



jedná o příjem nebo výdej. V dolní části se nachází zápatí, ve kterém se nacházejí navigační tlačítka, která jsou určena pro pohyb mezi položkami v rámci databáze.

V tomto formuláři se nachází podformulář, podle kterého uživatel může určit, v rámci jakého dodavatele a odběratele je transakce prováděna. Poslední položka podformuláře je *Název a Druh odpadu*, který náleží k dané transakci.

Pohyb transakcí

Přidat

Seznam

Hledat podle Transakce

Upravit

Tisk

Odstranit

Uložit do PDF

Hledat

Zrušit

ID Pohybu: PO567  
Datum: 31.03.2017  
Transakce: Příjem  
ID Dodavatele: DO123  
ID Odběratele: OD789  
ID Karty: KA789

Dodavatel			Odběratel			Odpad	
Název	Jméno	Příjmení	Název	Jméno	Příjmení	Název	Druh odpadu
Vitkovice steel				Miroslav	Onderka	Železo	železný odpad

⏮

⏪

⏩

⏭

**Obrázek 4-10:** Formulář Pohyb transakcí, zdroj: vlastní.

Pokud uživatel použije tlačítka *Přidat* nebo *Upravit*, tak se otevrou nové formuláře *frmPridatPohyb* a *frmUpravitPohyb*, podoba těchto formulářů je znázorněna na **Obrázku 4-11**. Skládají se celkem z šesti polí, do kterých uživatel může ukládat nebo upravovat důležitá data, která se týkají transakcí.

Přidání transakce	Upravení transakce
ID Pohybu	PO567
Datum	31.03.2017
Transakce	Příjem
ID Dodavatele	DO123
ID Odběratele	OD789
ID Karty	KA789
OK Storno	OK Storno

**Obrázek 4-11:** Formuláře přidat a upravit transakce, zdroj: vlastní.

#### 4.3.6 Formulář frmOdberatele

Formulář *frmOdberatele* slouží k evidenci veškerých odběratelů, kterým firma prodala jakékoliv množství železa, barevných kovů nebo papíru. Tento formulář je dostupný v navigačním pásu pod položkou *Odběratelé*.

Formulář obsahuje záhlaví, ve kterém je jeho název společně s tlačítky pro uživatelské ovládání. Při odstranění je uživatel dotázán, jestli chce skutečně položku odstranit, aby nedošlo k jejímu nechtěnému smazání. Zde je nutné zkontrolovat, zda tento odběratel není zadán ve formuláři *frmPohybTransakci*, protože by došlo k nekonzistenci dat a uživateli není povoleno tohoto odběratele odstranit. Následně se zde nachází vyhledávací pole, které slouží k vyhledávání určitého dodavatele podle jeho názvu v případě firmy, nebo podle jména, za předpokladu, že se jedná o fyzickou osobu.

Odběratelé		Přidat	Seznam	Hledat podle Názvu
		Upravit	Tisk	
		Odstranit	Uložit do PDF	Hledat Zrušit

ID Dodavatele	OD678
Název	Arcelor Mittal
Jméno	
Příjmení	
Číslo občanky	
Běžný účet	789789-7897897897/8978
IČO	76887897
DIČ	VZ7687897897
E-mail	info@arcelor.com
Mobil	+420 787 899 898
Město	Ostrava
Ulice	Vclovská
Číslo popisné	18
PSČ	710 00

◀ ▶ ⏮ ⏭

**Obrázek 4-12:** Formulář Odběratelé, zdroj: vlastní.

Tlačítka, která slouží pro přidání a upravení se odkazují na další formuláře *frmPridatOdberatele* a *frmUpravitOdberatele*, které jsou graficky a funkčně totožné s formuláři pro přidání a úpravu dodavatelů.

V dolní části se stejně jako v ostatních formulářích nacházejí navigační tlačítka pro přechod uživatele mezi jednotlivými položkami v databázi.

Jak je možno vidět na **Obrázku 4-12**, uživatelé mají možnost zobrazit seznam všech odběratelů, který lze následně vytisknout nebo lze uložit do souboru ve formátu PDF.

#### **4.3.7 Závěrečné přizpůsobení pro uživatele**

Při předání databázové aplikace uživatelům je převedena do uživatelského prostředí, ve kterém uživatel nemá přístup k jednotlivým databázovým objektům. To znamená, že uživatel ovládá aplikace pouze pomocí vytvořených formulářů a sestav, a to z důvodu, aby nemohl ohrozit funkcionality databázové aplikace. Při spuštění aplikace je nastaveno, aby se automaticky spustil domovský formulář *frmDomu*, ze kterého uživatel celou aplikaci ovládá. Ostatní ovládací nabídky a navigační tlačítka, která jsou součástí programu Microsoft Access jsou deaktivovány.

V období odevzdání této bakalářské práce bude databázová aplikace předána firmě k otestování v reálném provozu, kdy bude během několika prvních měsíců testována a při nalezení případných nedostatků budou tyto nedostatky odstraněny.

## 5 Závěr

Tato bakalářská práce byla realizována pro firmu Martin Trupar Upon. Cílem bylo vytvořit databázovou aplikaci, která bude navržena přímo pro konkrétní potřeby této firmy. Veškeré požadavky, které si firma stanovila, byly projednány s daným živnostníkem na několika schůzkách, na kterých došlo k prodiskutování a upřesnění stanovených podmínek.

Návrh aplikace probíhal pomocí programu Microsoft Access, ve kterém byly vytvořeny jednotlivé tabulky pro ukládání dat. Následně bylo vytvořeno uživatelské rozhraní, pomocí formulářů, kde pro správnou funkci a vzájemnou interakci byl použit programovací jazyk VBA (Visual Basic for Application). Nakonec byly vytvořeny sestavy pro tisk seznamů například dodavatelů a odpadů. Sestavy lze snadno vytisknout nebo uložit ve formátu PDF.

Cílem bakalářské práce bylo navrhnout a následně implementovat databázovou aplikaci, která plní následující funkce:

- jednoduché rozhraní a intuitivní ovládání,
- správa a záznam pohybu veškerého kovu a papíru,
- seznam všech údajů o dodavatelích a odběratelích,
- záznam veškerých skladů sběrný.

Všechny výše uvedené funkce byly při tvorbě databázové aplikace splněny. Aplikace bude nasazena ve firmě do provozu zároveň s odevzdáním bakalářské práce, jestliže ve zkušebním provozu nebudou nalezeny žádné nedostatky, tak aplikace přejde do ostrého provozu. V opačném případě budou nedostatky odstraněny.

V budoucnu by aplikace mohla být rozšířená o příjemku a výdejku odpadů, které jsou v databázi uloženy.

## Seznam použité literatury

### Tištěné zdroje:

BELKO, Peter. *Microsoft Access 2013: podrobná uživatelská příručka*. Brno: Computer Press, 2014. ISBN 978-80-251-4125-0.

CONNOLLY, T., C. BEGG a R HOLOWCZAK. *Mistrovství – databáze: profesionální průvodce tvorbou efektivních databází*. Brno: Computer Press, 2009. ISBN 978-80251-2328-7.

FARANA, Radim. *Databázové systémy*. Ostrava, 1995.

GÜNZL, Richard. *NoSQL databáze*. Praha, 2012. Bakalářská práce. Vysoká škola ekonomická v Praze.

OPPEL, Andrew J. *Databáze bez předchozích znalostí*. Brno: Computer Press, 2006. ISBN 80-251-1199-7.

*O Katalogu odpadů*. In: . 2016, ročník 2016, 93/2016 Sb.

SADALAGE, Pramod J. a Martin FOWLER. *NoSQL distilled: a brief guide to the emerging world of polyglot persistence*. Upper Saddle River, NJ: Addison-Wesley, c2013. ISBN 978-0-321-82662-6.

SHEPHER, Richard. *Access VBA: výukový průvodce*. Brno: Computer Press, 2012. ISBN 978-80-251-3686-7.

S. K. SINGH. *Database Systems: Concepts, Design and Applications*. Dorling Kindersley, 2011. ISBN 978-81-317-6092-5.

VIESCAS, John a Jeff CONRAD. *Mistrovství v Microsoft Office Access 2007*. Brno: Computer Press, 2008. Mistrovství. ISBN 978-80-251-2162-7.

### Elektronické zdroje:

*Access 2010: database tasks* [online]. Readmond: Microsoft, 2017 [cit. 2017-04-21]. Dostupné z: <https://support.office.com/en-us/article/Access-2010-database-tasks-268acfed-2484-4822-acb3-c30e58045588?omkt=en-US&ui=en-US&rs=en-US&ad=US>.

*Data types for Access desktop databases* [online]. Readmond: Microsoft, 2017 [cit. 2017-04-21]. Dostupné z: <https://support.office.com/en-us/article/Data-types-for-Access-desktop-databases-df2b83ba-cef6-436d-b679-3418f622e482?omkt=en-US&ui=en-US&rs=en-US&ad=US>.

*Description of the database normalization basics* [online]. Readmond: Microsoft, 2017 [cit. 2017-04-21]. Dostupné z: <https://support.microsoft.com/en-us/help/283878/description-of-the-database-normalization-basics>.

*Guide to table relationships* [online]. Readmond: Microsoft, 2017 [cit. 2017-04-21]. Dostupné z: <https://support.office.com/en-gb/article/Guide-to-table-relationships-30446197-4fbe-457b-b992-2f6fb812b58f?fromAR=1&omkt=en-GB&ui=en-US&rs=en-GB&ad=GB>.

HAUZAR, David. *Tvorba databází v MySQL – I. In: AbcLinuxu* [online]. 20.03.2003 [cit. 2017-04-21]. Dostupné z: <http://www.abclinuxu.cz/clanky/navody/tvorba-databazi-v-mysql-i>.

*Introduction to reports in Access* [online]. Readmond: Microsoft, 2017 [cit. 2017-04-21]. Dostupné z: <https://support.office.com/en-us/article/Introduction-to-reports-in-Access-e0869f59-7536-4d19-8e05-7158dcd3681c?omkt=en-US&ui=en-US&rs=en-US&ad=US>.

*Inisoft.cz: Postup a doporučení pro splnění ohlašovacích povinností v oblasti odpadového hospodářství v roce 2017* [online]. Inisoft.cz, 2017 [cit. 2017-04-21]. Dostupné z: <https://www.inisoft.cz/ispop>.

Podnikatel.cz: *Zákon o odpadech – Sběr a výkup odpadů* [online]. Podnikatel.cz, 2001 [cit. 2017-04-21]. Dostupné z: <http://www.podnikatel.cz/zakony/zakon-c-185-2001-sb-zakon-o-odpadech/f2197868/#f2200054>.

## Seznam zkratek

1NF – První normální forma

2NF – Druhá normální forma

3NF – Třetí normální forma

4NF – Čtvrtá normální forma

5NF – Pátá normální forma

ACID – Atomicity, Consistency, Isolation, Durability

BCNF – Boyce/Coddova normální forma

BU – Běžný účet

ČR – Česká republika

ČO – Číslo občanských průkazů

DIČ – Daňové identifikační číslo

GQL – Google Query Language

HQL – Hypertext Query Language

IČO – Identifikační číslo organizace

ID – Identifikace

IDMS – Integrated Database Management System

IMS – Information Management Systém

PDF – Portable Document Format

RDBMS – Relational database management system

SQL – Structured Query Language

URL – Uniform Resource Locator

VBA – Visual Basic for Applications

WWW – World Wide Web

# Prohlášení o využití bakalářské práce

Prohlašuji, že

- byl jsem seznámen s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo;
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě bakalářskou práci užít (§ 35 odst. 3);
- souhlasím s tím, že bakalářská práce bude v elektronické podobě archivována v Ústřední knihovně VŠB-TUO a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že bibliografické údaje o bakalářské práci budou zveřejněny v informačním systému VŠB-TUO;
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona;
- bylo sjednáno, že užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne 3. 5. 2017



.....

jméno a příjmení studenta



## **Seznam příloh**

Příloha č. 1 – ukázka seznamu dodavatelů

Příloha č. 2 – ukázka seznamu skladů

Příloha č. 3 – ukázka seznamu odpadů

Příloha č. 4 – CD s databázovou aplikací

## Přílohy

### Příloha č. 1

#### Dodavatelé

Název		Jméno		Příjmení		Číslo občanky	Běžný účet
Mittal steel							787779-890896745
IČO	DIČ	E-mail		Mobil		Město	
67545607	VB6567678678	info@mittal.com		+420 878 987 897		Ostrava	
Ulice		Číslo popisné	PSČ				
Vítkovická		6778	710 00				

Název		Jméno		Příjmení		Číslo občanky	Běžný účet
Vítkovice steel							789789-879089089
IČO	DIČ	E-mail		Mobil		Město	
67868678	GH7678686786	info@vs.cz		+420 609 676 575		Ostrava	
Ulice		Číslo popisné	PSČ				
Podružková		30	710 00				

Název		Jméno		Příjmení		Číslo občanky	Běžný účet
		Lukáš		Kubečka		878989089	456768-979890809
IČO	DIČ	E-mail		Mobil		Město	
		kubeka@gmail.com		+420 609 875 675		Opava	
Ulice		Číslo popisné	PSČ				
Pekařská		20	746 01				

Název		Jméno		Příjmení		Číslo občanky	Běžný účet
		Michal		Adámek		787897897	000000-767867657
IČO	DIČ	E-mail		Mobil		Město	
		adamek@gmail.com		+420 768 897 898		Olomouc	
Ulice		Číslo popisné	PSČ				
Brněnská		6787	567 56				

Název		Jméno		Příjmení		Číslo občanky	Běžný účet
Ostroj Opava							897897-897897897
IČO	DIČ	E-mail		Mobil		Město	
76756456	HJ5675675656	info@ostroj.com		+420 987 897 979		Opava	
Ulice		Číslo popisné	PSČ				
Těšínská		111	746 01				

## Příloha č. 2

### Sklady

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Měď	Opava	Myslbekova	2	746 01

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Cín	Ostrava	Těšínská	567	710 00

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Stroje	Opava	Myslbekova	2	746 01

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Ocel	Opava	Myslbekova	2	746 01

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Nerez	Ostrava	Vdovská	18	710 00

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Papír	Ostrava	Těšínská	567	710 00

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Nářadí	Opava	Myslbekova	2	746 01

Druh skladu	Město	Ulice	Číslo popisné	PSČ
Železo	Ostrava	Vdovská	18	710 00

### Příloha č. 3

#### Karta odpadů

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Barevné kovy	3000	kg	Hliník	1744-02	30,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Nebezpečný odpad	40	kg	Autobaterie	175092*	5,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Papír	67	kg	Lepenka	1501-01	5,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Barevné kovy	1000	kg	Mosaz	1704-01	50,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Papír	2000	kg	Časopisy	1501-02	2,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Železo	50	kg	Železný odpad	1744-05	20,00 Kč

Název	Množství	Jednotky	Druh odpadu	Kód odpadu	Cena
Barevné kovy	70	kg	Měď	1704-01	200,00 Kč